York University
Dept. of Electrical Engineering and Computer Science

SPICE Tutorial

EECS2200

Fall 2015-2016

I. SPICE Tutorial

Introduction:

SPICE (Simulation Program with Integrated Circuits Emphasis) is a general-purpose open source analog electric circuit simulator. It started at the University of California Berkeley [1-3] and many implementations exist now, but all are based on the original Berkeley SPICE. You can download a Spice simulator from: http://ngspice.sourceforge.net/download.html.

Spice needs a description of the circuit (*netlist*) to run the simulation and commands on what exactly to do. It performs the simulation and displays the results. SPICE can simulate many circuits with elements such as resistors, capacitors, inductors, transmission lines, many different types of transistors, etc.

As mentioned above, there are many versions of SPICE. Although they all share the basic idea, some difference mays exist between the versions. In this tutorial we will concentrate on two versions of SPICE, i.e.

1. ngspice: a GNU simulator
2. HSPICE: ORCAD simulator.

If there are differences, it will be mentioned in the context.

To run SPICE, you need to supply a netlist and control commands.

Netlist: the netlist of a circuit is a description of the circuits, the circuit elements and how they are connected. There are two ways to do this. The first is to use SPICE language to describe the circuit. The second is to use a schematic capture program to generate the netlist. A schematic capture program is a GUI interface that lets you draw the circuit, then it transform the circuit you draw into a netlist that could be used by SPICE.

The simulation engine itself can be an interactive program, that reads the netlist, and then you provide commands to the program on what you want to do and display the simulation results (ngspice). Or, the engine could be integrated with the schematic capture, so you draw the circuit, choose the simulation profile, and perform the simulation (HSPICE). It is important that you know both techniques.

This tutorial is a very basic introduction to SPICE. It covers only the parts necessary to use in this course. As you progress in your study, especially if you consider a hardware related career, you will be introduced more and more to SPICE and some other CAD tools.

SPICE Input File Syntax

The SPICE program consists of three parts:
Data Statement: describes the circuit to simulate
Control Statement: tells SPICE what type of analysis to perform
Output statement; specifies the output to be displayed
For example in ngspice, the input file consists of the netlist, that is the first part describing the circuit. Control and output display is performed interactively after the circuit is processed.

A general description of the input file.

- The first line is the title of the circuit. It will be displayed with the results.
- Any line starts with (*) is a comment line, will be ignored by the simulator
- Any line starts with (+) is a continuation line. That means it is a part (continuation) of the previous line.
- Control and command lines starts with a (.).

Netlist

All nodes must have labels (positive integer is compatible with all versions of SPICE).
There must be a ground node (node 0).
All circuit elements must have a text label; the first letter of the label specifies the type of the circuit element. Here is a partial list

V	Independent voltage source
I	Independent current source
R	Resistor
C	Capacitor
L	Inductor
D	Diode
E	Voltage controlled voltage source
G	Voltage controlled current source
H	Current controlled voltage source
F	Current controlled current source

Each element is described by the element name, the two nodes it is connected to, and value.

Example:
RES1 1 0 3500

This is a resistor labelled RES1 connected between nodes 1 and 0 and have a value of 3500 ohms. There could be more optional fields, but will not cover them now.

The value could be written as
3500, 3.5E3 3.5K 0.0035M, etc.

Where T is tera, G is giga, M is Mega, K is kilo, m is milli, u is micro, n is nano, p is pita, and f is femto. Original SPICE accepts only capital letters, recent versions accepts both small and capital letters.

Example:
CIN1 3 0 47PF <IN>
The previous statement describes a capacitor connecting nodes 3 and 0 with a value of 47 Pico Farads. <IN> is an optional initial condition. The first node 3 is the positive node when calculating the voltage on the capacitor.

Voltage and current sources:

DC voltage
The DC voltage is defined as: Vname n1 n2 Type value

Example:
Vin n5 n7 DC 10

A voltage source called Vin between nodes n1 (+ve) and n2 (-ve) DC source with 10Volts
Note that the current flows from the positive node to the negative node inside the source.

For sinusoidal sources: Vname n1 n2 SIN(V0 VA f t_D Theta Phase)
where

$$V_{name} = V_0 + V_A\, e^{-\theta(t-t_D)} \sin\left[2\pi f(t-t_D) + (Phase/360)\right]$$

V0 = offset voltage in volt
VA = amplitude in volt
f = frequency in Hz
Theta = damping factor per second
Phase = phase in degrees

Vsin 1 2 SIN(0 5 50)
is a sine wave with 0 offset, magnitude of 5 and frequency of 50 Hz (since Td, theta and phase are zeros, we did not include them)

Commands (Control)

.OP
.OP stands for operating point, just calculate DC operating point of the circuit. Some versions of SPICE will do this always, some need to be instructed to do so by using the .OP

DC sweep analysis
The command .DC changes the specified independent source (voltage or current) over the interval specified by the user and calculates the OP analysis

Example:
.DC vin 0.25 5.0 0.25
Changes the value of the independent voltage source vin from 0.25V to 5.0V in increments of 0.25V

AC sweep analysis
In AC sweep analysis, we change the frequency of all AC sources in the circuit according to the control statement
.AC increment_type number_of_points start end

Examples:
.AC DEC 10 100 10KHz
Performs an AC sweep, DEC stands for decade, 10 points per decade, starting at 100HZ and ending at 100KHz.

.AC OCT 5 100 100KHz
Oct for octave (a factor of 8) with 5 points per octave, starting at 100 ending at 100KHz.

.AC LIN 50 100 100KHz
LIN for linear, with 50 points in total starting from 100Hz to 100KHz.

Transient analysis
.TRAN Tstep Tstop [Tstart [Tmax]] [UIC]
Tstep is the time step in calculating the transient response. Tstop is the end time.
Tstart is an optional for the start time, if not mentioned it is zero.

Tmax is the maximum step size that ngspice uses;
`UIC` (use initial conditions) is an optional keyword which indicates that the user does not want ngspice to solve for the quiescent operating point before beginning the transient analysis. If this keyword is specified, ngspice uses the values specified in the circuit description.

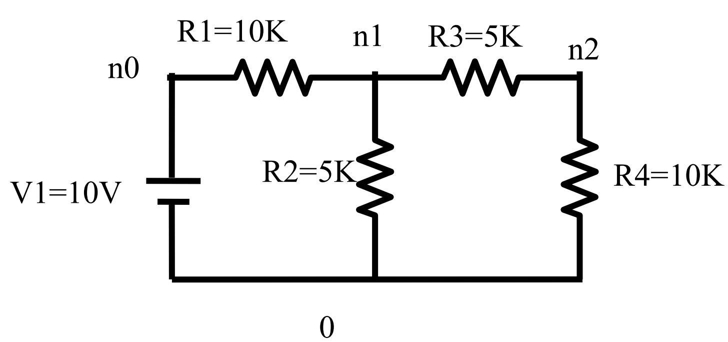
Output statements
.PRINT and .PLOT are used to print and plot some values. The examples given below will illustrate this point.
The control statement, could be included in a .control .endc block in the netlist file. In ngspice if included in that block, do not use capital letters or "." before the command

Here is a simple example using ngspice, that netlist describes the circuit shown below

Here is a file called ex1.net

```
Example 1
V1 n0 0 DC 10
R1 n0 n1 10K
R2 n1 0 5K
R3 n1 n2 5K
R4 n2 0 10K
.END
```

Save this program in a file called ex1.net



Start ngspice

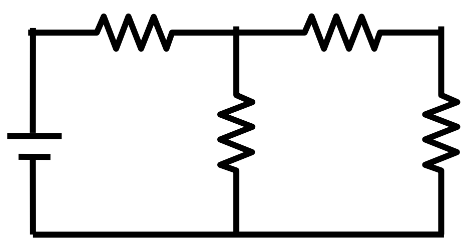
```
tigger 136 % /tmp/cse/local/bin/ngspice
******
** ngspice-24 : Circuit level simulation program
** The U. C. Berkeley CAD Group
** Copyright 1985-1994, Regents of the University of California.

** Please get your ngspice manual from http://ngspice.sourceforge.net/docs.html

** Please file your bug-reports at http://ngspice.sourceforge.net/bugrep.html

** Creation Date: Tue Jul 17 12:01:33 EDT 2012

******

ngspice 115 -> source ex1.net

Circuit: example 1

ngspice 116 -> op
Doing analysis at TEMP = 27.000000 and TNOM = 27.000000

No. of Data Rows : 1

ngspice 117 -> print n1
n1 = 2.727273e+00
ngspice 118 -> print n2
n2 = 1.818182e+00
ngspice 119 -> print i(v1)
i(v1) = -7.27273e-04
ngspice 120 ->
```

Note that in ngspice you can print the currents through a voltage sources only. However you can insert a zero voltage source in series with any load and you can print the current through that voltage source (same current in the load).

The control statement we entered (shown in red) can be included in the netlist file. This is included between .control and .endc statements.
Write this file and store it in ex2.net

Note that the “dc V1 0.0 5.0 0.5” sweeps the voltage source V1 from 0 to 5V with an increment of 0.5 and the print v(2) prints the voltage at v2 for every value of V1.

```
Example 2
V1 1 0 DC 10
R1 1 2 10K
R2 2 0 5K
R3 2 3  5K
R4 3 0 10K
.END
.control
dc v1 0.0 5.0 0.5
print v(2)
.endc
```



Run ngspice. The figure below shows the result

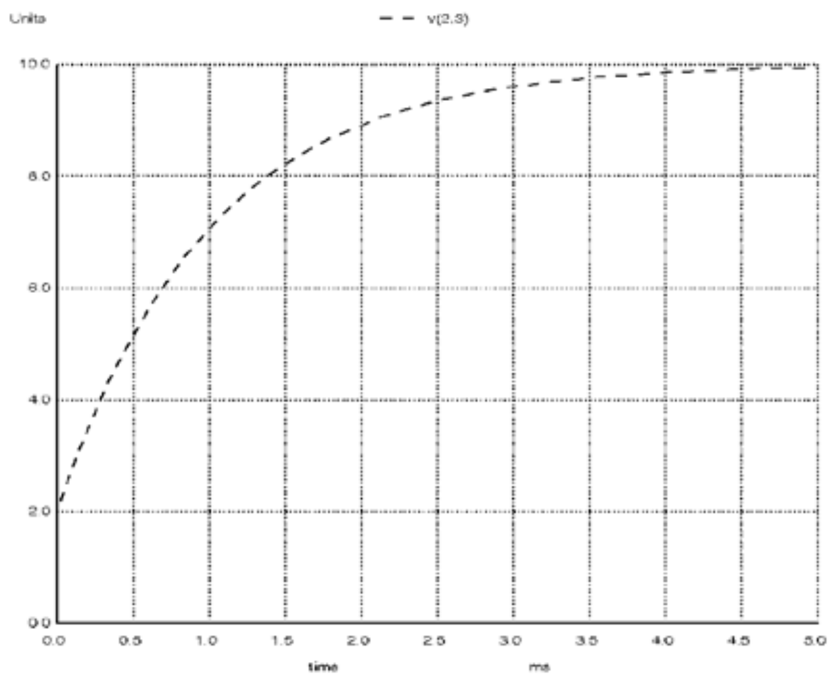
```
tigger 57 % ngspice
******
** ngspice-24 : Circuit level simulation program
** The U. C. Berkeley CAD Group
** Copyright 1985-1994, Regents of the University of California.
** Please get your ngspice manual from
http://ngspice.sourceforge.net/docs.html
** Please file your bug-reports at http://ngspice.sourceforge.net/bugrep.html
** Creation Date: Thu Sep  6 12:00:52 EDT 2012
******
ngspice 313 -> source ex2.net

Circuit: example 2

Doing analysis at TEMP = 27.000000 and TNOM = 27.000000



No. of Data Rows : 11
   example 2
   DC transfer characteristic  Fri Sep 14 10:00:35  2012
--------------------------------------------------------------------------------
Index   v-sweep        v(2)
--------------------------------------------------------------------------------
0         0.000000e+00 0.000000e+00
1         5.000000e-01  1.363636e-01
2         1.000000e+00 2.727273e-01
3         1.500000e+00 4.090909e-01
4         2.000000e+00 5.454545e-01
5         2.500000e+00 6.818182e-01
6         3.000000e+00 8.181818e-01
7         3.500000e+00 9.545455e-01
8         4.000000e+00 1.090909e+00
9         4.500000e+00 1.227273e+00
10       5.000000e+00 1.363636e+00
ngspice 314 ->
```

Next, we will simulate an RC circuit. Write the following file and store it in a file named ex_rc.net. Run ngspice, source the file and run it.

```
Simple example for spice3 tutorial
*Sources
V1 1 0 DC 10V
*Network Elements
R1 1 2 500
C1 2 3 1uF IC=2V
R2 3 0 500
.END
.control
run
tran 0.1ms 5ms UIC
plot v(2,3)
.endc
```

Screen capture of ngspice

```
ngspice 314 -> source ex5.net

Circuit: simple example for spice3 tutorial

ngspice 315 -> run
Doing analysis at TEMP = 27.000000 and TNOM = 27.000000

ngspice 316 -> source ex_rc.net

Circuit: simple example for spice3 tutorial

Doing analysis at TEMP = 27.000000 and TNOM = 27.000000

Doing analysis at TEMP = 27.000000 and TNOM = 27.000000


Initial Transient Solution
--------------------------

Node                          Voltage
----                          -------
1                               0
2                               0
3                               0
v1#branch                       0

No. of Data Rows : 59
Warning: Missing charsets in String to FontSet conversion
Warning: Unable to load any usable fontset
ngspice 317 ->
The file "/tmp/hc600" may be printed on a postscript printer.

ngspice 317 ->
```



Reference

1. Nagel, L. W, and Pederson, D. O., *SPICE (Simulation Program with Integrated Circuit Emphasis)*, Memorandum No. ERL-M382, University of California, Berkeley, Apr. 1973.
2. Nagel, Laurence W., *SPICE2: A Computer Program to Simulate Semiconductor Circuits*, Memorandum No. ERL-M520, University of California, Berkeley, May 1975.
3. http://ngspice.sourceforge.net/docs.html.